DTU Library

Penalty parameter of the penalty function method

Si, Cheng Yong; Lan, Tian; Hu, Junjie; Wang, Lei; Wu, Qi Di

Published in:
Kongzhi yu Juece

Link to article, DOI:
10.13195/j.kzyjc.2013.0640

Publication date:
2014

Document Version
Publisher's PDF, also known as Version of record

Citation (APA):
Si, C. Y., Lan, T., Hu, J., Wang, L., & Wu, Q. D. (2014). Penalty parameter of the penalty function method. *Kongzhi yu Juece*, *29*(9), 1707-1710. https://doi.org/10.13195/j.kzyjc.2013.0640

第 29 卷 第 9 期
Vol. 29 No. 9

控 制 与 决 策
Control and Decision

2014 年 9 月
Sep. 2014

文章编号: 1001-0920 (2014) 09-1707-04

DOI: 10.13195/j.kzyjc.2013.0640

关于惩罚函数中惩罚系数的讨论

司呈勇[1], 兰 天[2], 胡俊杰[3], 汪 镭[1], 吴启迪[1]

(1. 同济大学 电子与信息工程学院，上海 201804；2. 柏林工业大学 新能源电网研究所，柏林 10587；3. 丹麦科技大学 电气工程系，哥本哈根 2800)

摘 要: 对惩罚函数法中的惩罚系数进行了系统分析和讨论. 首先, 针对 Deb 基于可行性规则对约束违反相同情况下的比较没有具体说明的现状, 提出一种改进的 Deb 基于可行性规则. 在此基础上, 证明了惩罚系数过大或者过小均不会影响排序的结论, 并给出了惩罚系数影响排序的上下边界. 实例分析表明了所得结论的有效性, 为基于惩罚系数的算法设计提供了依据.

关键词: 惩罚系数；Deb 基于可行性规则；约束处理技术；排序策略

中图分类号: TP18 **文献标志码:** A

Penalty parameter of the penalty function method

SI Cheng-yong[1], *LAN Tian*[2], *HU Jun-jie*[3], *WANG Lei*[1], *WU Qi-di*[1]

(1. College of Electronics and Information Engineering，Tongji University，Shanghai 201804，China；2. Institute of Sustainable Electric Networks and Sources of Energy，Technical University of Berlin，Berlin 10587，Germany；3. Department of Electrical Engineering，Technical University of Denmark，Copenhagen 2800，Denmark. Correspondent：WANG Lei，E-mail：wanglei@tongji.edu.cn)

Abstract: The penalty parameter of penalty function method is systematically analyzed and discussed. For the problem that Deb's feasibility-based rule doesnot give the detailed instruction as how to rank two solutions when they have the same constraint violation, an improved Deb's feasibility-based rule is presented, which can be seen as a reference standard. And based on this, the upper and lower boundary of penalty parameter that affects the ranking is obtained. The example verifies the effectiveness of the systematical analysis, which provides a basis for the future algorithm design based on the penalty parameters.

Key words: penalty parameter；Deb's feasibility-based rule；constraint handling techniques；ranking methods

0 引 言

约束优化问题是许多实际工程中面临的一类较难求解的问题, 对其探究具有重要的理论和实际意义. 进化算法, 作为一种模拟自然进化过程的优化方法, 以其较好的适应性, 在自动控制、函数优化、生产调度、管理科学等许多领域得到了广泛应用. 考虑到进化算法本质上是一种无约束搜索技术和用来产生解的策略, 若将其用于求解约束优化问题, 则需要额外的机制来处理约束. 相应地, 许多约束优化进化算法相继涌现[1-4].

惩罚函数法, 作为处理约束最常见的方法之一, 利用惩罚系数对目标函数值和约束违反值进行平衡, 并对进化算法的解进行排序. 利用惩罚函数法的关键和难点是惩罚系数的确定. 目前, 主要通过实验来确定“相对”最优的惩罚系数. 事实上, 惩罚系数过大或过小对排序均不会产生影响. 此时, 对惩罚系数的加大或者减小都是无意义的. 所以对惩罚系数的系统分析, 不仅可以减少很多不必要的盲目试凑, 而且可以为进一步设计适应性惩罚函数法提供依据.

针对上面所述, 本文对惩罚系数的内部机理进行系统探讨. 主要思路是以 Deb 基于可行性规则作为参照标准, 由根据规则排序的唯一性得出惩罚系数影响排序的上下边界, 说明其设置过大或过小对排序结果均无影响. 最后通过实例表明了以上分析的有效性.

收稿日期: 2013-05-17；**修回日期:** 2013-09-16.

基金项目: 国家自然科学基金项目(61075064, 61034004, 61005090)；教育部新世纪人才计划项目(NCET-10-0633)；上海市金融信息技术研究重点实验室开放课题基金项目.

作者简介: 司呈勇(1986−), 男, 博士生, 从事群体智能的研究；汪镭(1970−), 男, 教授, 博士生导师, 从事群体智能、智能控制等研究.

1 约束处理技术

1.1 约束优化问题描述

不失一般性, 考虑如下形式的约束优化问题:

$$\begin{aligned} &\min\ f(\boldsymbol{x}):\ \boldsymbol{x}=(x_1,x_2,\cdots,x_n).\\ &\text{s.t.}\ \ g_j(\boldsymbol{x})\leqslant 0,\ j=1,\cdots,l;\\ &\qquad h_j(\boldsymbol{x})=0,\ j=l+1,\cdots,m;\\ &\qquad l(i)\leqslant x_i\leqslant u(i),\ i=1,\cdots,n. \end{aligned} \tag{1}$$

其中: $f(\boldsymbol{x})$ 为目标函数, $g_j(\boldsymbol{x})$ 和 $h_j(\boldsymbol{x})$ 分别为不等式约束条件和等式约束条件.

1.2 约束处理

惩罚函数法是处理约束应用最为广泛的一种方法, 其主要思想是在目标函数的基础上增加惩罚项, 从而在约束空间内进行有偏搜索[5].

由于惩罚函数法对惩罚系数的依赖性, 一些不需要参数, 而是基于区分可行解和不可行解的方法相继提出. 例如, Deb[6] 从解的可行性角度提出一种准则来两两比较个体: 1) 可行个体与不可行个体比较, 可行个体较优; 2) 两个可行个体比较, 目标函数值较小的个体较优; 3) 两个不可行个体比较, 违反约束条件程度小的个体较优.

1.3 Deb 基于可行性规则的改进

根据 Deb 基于可行性规则, 如果两个不可行解具有相同的约束违反, 则它们具有相同的性能. 这种情况下, 它们将根据上一代的位置 (产生时间) 进行排序, 即上一代中排在前面的一代仍然排在前面. 可以看出, 这种排序与实际的关系并不一致.

例如, 假设有一个解 $\boldsymbol{f}(\boldsymbol{x})=(\boldsymbol{f}_1(\boldsymbol{x}),\boldsymbol{f}_2(\boldsymbol{x}))=(f(\boldsymbol{x}),G(\boldsymbol{x}))$, 其中括号中的第 1 个数为解的目标函数值, 第 2 个数代表这个解违反约束的程度. 问题为求解最小化.

假设其中有一组解为

$$S=\{(9,3),(4,0),(2,3),(7,3),(2,0),(1,1)\},$$

根据 Deb 基于可行性规则排序后的结果为

$$S'=\{(2,0),(4,0),(1,1),(9,3),(2,3),(7,3)\}.$$

可以看出, 解 $(9,3)$ 排在解 $(2,3)$ 和 $(7,3)$ 之前, 尽管它的性能不如后面两个 (目标函数值更大). 这主要是因为在原来的序列 S 中, 与解 $(2,3)$ 和 $(7,3)$ 相比, 解 $(9,3)$ 排在更前面.

为了克服这个不足, 本文在 Deb 基于可行性规则之后增加一条规则: 如果两个不可行解具有相同的约束违反, 则根据目标函数值对它们进行排序.

采用改进后的规则排序后的结果为

$$S'=\{(2,0),(4,0),(1,1),(2,3),(7,3)(9,3)\}.$$

尽管这只是一个小的改进, 但它是非常有用的, 特别是在进化过程的开始阶段.

2 惩罚系数的系统分析

惩罚系数的引入可使一个约束优化问题 (A) 转化为一个无约束优化问题 (A')[7]. 这里, 将评价函数 L 定义如下.

对于给定的 $\lambda,\delta>0$, 设

$$L(\boldsymbol{x}_i,\lambda,\delta)=f(\boldsymbol{x}_i)+\lambda\cdot G(\boldsymbol{x}_i,\delta),\ i=1,2,\cdots,\mathrm{NP}. \tag{2}$$

其中: $\boldsymbol{x}_i$ 为种群中 NP 个 n 维实值向量, λ 为惩罚系数, δ 为等式约束条件的容忍值, f 和 G 分别为目标函数和约束条件. 这里

$$G(\boldsymbol{x}_i,\delta)=\sum_{j=1}^{m}G_j(\boldsymbol{x}_i,\delta)=$$

$$\sum_{j=1}^{l}\max(0,g_j(\boldsymbol{x}_i))+\sum_{j=l+1}^{m}\max(0,|h_j(\boldsymbol{x}_i)|-\delta). \tag{3}$$

由于本文主要考虑 λ 的作用, 可以将 δ 设为常数, 则式 (2) 变为

$$L(\boldsymbol{x}_i,\lambda)=f(\boldsymbol{x}_i)+\lambda\cdot G(\boldsymbol{x}_i),\ i=1,2,\cdots,\mathrm{NP}. \tag{4}$$

如文献 [8] 所述, 一般情况下, 在进化过程中, 群体将不可避免地出现以下 3 种情形: 1) 仅包含不可行个体 (不可行情形); 2) 同时包含不可行个体和可行个体 (半可行情形); 3) 全为可行个体 (可行情形).

由于惩罚系数 λ 不会影响可行情形下的评价函数 L, 这里仅对不可行和半可行情形加以讨论.

给定两个种群个体 $\boldsymbol{x}_s$ 和 $\boldsymbol{x}_t$, 其中 s 和 t 在区间 $[1,\mathrm{NP}]$ 中随机选择, 并满足 $s\neq t$. 这两个个体的评价函数值之差为

$$\begin{aligned} &\Delta(\boldsymbol{x}_s,\boldsymbol{x}_t,\lambda)=L(\boldsymbol{x}_s,\lambda)-L(\boldsymbol{x}_t,\lambda)=\\ &[f(\boldsymbol{x}_s)+\lambda\cdot G(\boldsymbol{x}_s)]-[f(\boldsymbol{x}_t)+\lambda\cdot G(\boldsymbol{x}_t)]=\\ &[f(\boldsymbol{x}_s)-f(\boldsymbol{x}_t)]+\lambda\cdot[G(\boldsymbol{x}_s)-G(\boldsymbol{x}_t)]. \end{aligned} \tag{5}$$

定义 $\Delta f_{st}=f(\boldsymbol{x}_s)-f(\boldsymbol{x}_t),\Delta G_{st}=G(\boldsymbol{x}_s)-G(\boldsymbol{x}_t)$, 则式 (5) 可写为

$$\Delta(\boldsymbol{x}_s,\boldsymbol{x}_t,\lambda)=\Delta f_{st}+\lambda\cdot\Delta G_{st}. \tag{6}$$

这里, 引入改进的 Deb 基于可行性规则作为参照标准, 并将其 3 条规则按可行解和不可行解情况进行重新组合, 即可行解优于/劣于不可行解; 不可行解按照违反约束条件程度/违反约束条件程度的倒数进行排序. 若根据 $\Delta(\boldsymbol{x}_s,\boldsymbol{x}_t,\lambda)$ 与根据以上 4 种组合得到的排序结果完全一致, 则说明排序结果是唯一的, 排序不依赖于惩罚系数.

2.1 不可行情形

根据 Deb 基于可行性规则, 两个不可行个体的排

序将按照它们的约束违反情况进行. 所以, 当 $\Delta(\boldsymbol{x}_s, \boldsymbol{x}_t, \lambda)$ 和 ΔG_{st} 具有相同符号时, 说明这两种方法在对这两个个体进行排序时具有相同效果.

在群体不可行情形下, 有 3 种不同的情况:

1) $\Delta G_{st} > 0$: 此时, 如果 $\lambda > -\Delta f_{st}/\Delta G_{st}$, 则 $\Delta(\boldsymbol{x}_s, \boldsymbol{x}_t, \lambda) = \Delta f_{st} + \lambda \cdot \Delta G_{st} > 0$;

2) $\Delta G_{st} < 0$: 此时, 如果 $\lambda > -\Delta f_{st}/\Delta G_{st}$, 则 $\Delta(\boldsymbol{x}_s, \boldsymbol{x}_t, \lambda) = \Delta f_{st} + \lambda \cdot \Delta G_{st} < 0$;

3) $\Delta G_{st} = 0$: 此时, 两个个体具有相同的约束违反, 所以 $\Delta(\boldsymbol{x}_s, \boldsymbol{x}_t, \lambda) = \Delta f_{st}$, 与 λ 无关. 根据前面改进的 Deb 基于可行性规则, 在违反约束相同的情况下, 根据目标函数值确定排序.

总之, 当对两个不可行个体 $\boldsymbol{x}_s$ 和 $\boldsymbol{x}_t$ 进行排序时, 如果 $\lambda > -\Delta f_{st}/\Delta G_{st}$, $\Delta G_{st} \neq 0$, 则这两种方法具有相同的排序效果. 这个结论可以扩展到整个种群.

定义 $\lambda_{\max} = \max(-\Delta f_{st}/\Delta G_{st})$, $\lambda_{\min} = \min(-\Delta f_{st}/\Delta G_{st})$, $i = 1, 2, \cdots, \mathrm{NP}$, $j = 1, 2, \cdots, \mathrm{NP}$, 且 $G(i) \neq G(j)$. 可得如下结论: 在不可行情形下, 对于整个种群排序, 如果 $\lambda > \lambda_{\max}$, 则这两种方法具有相同的效果; 如果 $\lambda_{\min} \leqslant \lambda \leqslant \lambda_{\max}$, 则两种方法的排序结果将会部分相同; 如果 $\lambda < \lambda_{\min}$, 则两种方法的排序结果完全相反, 惩罚函数法会根据违反约束的倒数进行排序 (具有最大违反约束的个体反而会排名第 1).

综上, 在 $\lambda > \lambda_{\max}$ 或者 $\lambda < \lambda_{\min}$ 的情况下, 种群排序不再依赖于惩罚系数 λ.

2.2　半可行情形

假设种群正经历半可行情形, 且种群中的可行和不可行个体数目分别为 p 和 q, 其中 $1 \leqslant p \leqslant \mathrm{NP}$, $1 \leqslant q \leqslant \mathrm{NP}$ 且 $p + q = \mathrm{NP}$.

1) 不可行个体与可行个体: 类似地, 给定两个种群个体 $\boldsymbol{x}_s$ 和 $\boldsymbol{x}_t$, 其中 s 和 t 分别为从 p 个可行个体和 q 个不可行个体中随机选择的个体, 则式 (5) 变为

$$\begin{aligned}&\Delta(\boldsymbol{x}_s, \boldsymbol{x}_t, \lambda) = L(\boldsymbol{x}_s, \lambda) - L(\boldsymbol{x}_t, \lambda) = \\ &f(\boldsymbol{x}_s) - [f(\boldsymbol{x}_t) + \lambda \cdot G(\boldsymbol{x}_t)] = \\ &[f(\boldsymbol{x}_s) - f(\boldsymbol{x}_t)] - \lambda \cdot G(\boldsymbol{x}_t).\end{aligned} \tag{7}$$

根据 Deb 基于可行性规则, 个体 $\boldsymbol{x}_s$ 优于个体 $\boldsymbol{x}_t$. 根据式 (7), 如果 $\lambda > \Delta f_{st}/G(\boldsymbol{x}_t)$, 则 $\Delta(\boldsymbol{x}_s, \boldsymbol{x}_t, \lambda) < 0$. 这种情况下, 在对这两个个体进行排序时, Deb 基于可行性规则与惩罚函数法具有相同的效果.

2) 不可行个体与不可行个体: 这种比较已在 2.1 节中进行了讨论. 当 $\lambda > -\Delta f_{st}/\Delta G_{st}$ 时 (这里, s 和 t 从 q 个不可行个体中随机选择), 这两种方法具有相同的效果.

记 I_{fea} 和 I_{inf} 分别为 p 个可行个体和 q 个不可行个体的索引集合. 假设

$$\begin{aligned}S_{\mathrm{inf}} = \Big\{&\lambda_{ij} | \lambda_{ij} = -\frac{\Delta f_{st}}{\Delta G_{st}},\ i = I_{\mathrm{inf}}(1), \cdots, I_{\mathrm{inf}}(q);\\ &j = I_{\mathrm{inf}}(1), \cdots, I_{\mathrm{inf}}(q)\ 且\ G(i) \neq G(j)\Big\},\\ S_{\mathrm{sem}} = \Big\{&\lambda_{ij} | \lambda_{ij} = \frac{\Delta f_{st}}{G(\boldsymbol{x}_t)},\ i = I_{\mathrm{fea}}(1), \cdots, I_{\mathrm{fea}}(p);\\ &j = I_{\mathrm{inf}}(1), \cdots, I_{\mathrm{inf}}(q)\Big\},\end{aligned}$$

则 λ 的集合可以描述为 $S = \{S_{\mathrm{inf}}, S_{\mathrm{sem}}\}$. 对于 S_{inf}, 定义 $\lambda_{\mathrm{inf}}^{\max} = \max(\lambda_{ij})$, $\lambda_{\mathrm{inf}}^{\min} = \min(\lambda_{ij})$, 其中 $i = I_{\mathrm{inf}}(1), \cdots, I_{\mathrm{inf}}(q)$, $j = I_{\mathrm{inf}}(1), \cdots, I_{\mathrm{inf}}(q)$ 且 $G(i) \neq G(j)$. 类似地, 对于 S_{sem}, 定义 $\lambda_{\mathrm{sem}}^{\max} = \max(\lambda_{ij})$, $\lambda_{\mathrm{sem}}^{\min} = \min(\lambda_{ij})$. 其中: $i = I_{\mathrm{fea}}(1), \cdots, I_{\mathrm{fea}}(p)$, $j = I_{\mathrm{inf}}(1), \cdots, I_{\mathrm{inf}}(q)$. 则 S 中的 λ 最大值 $\lambda_{\max} = \max(\lambda_{\mathrm{inf}}^{\max}, \lambda_{\mathrm{sem}}^{\max})$, λ 最小值 $\lambda_{\min} = \min(\lambda_{\mathrm{inf}}^{\min}, \lambda_{\mathrm{sem}}^{\min})$.

这种情形下, 有如下 5 种不同的情况.

① $\lambda > \lambda_{\max}$. 在对整个种群排序时, 两种方法具有相同的排序效果.

② $\lambda_{\mathrm{sem}}^{\max} < \lambda \leqslant \lambda_{\max}$. 根据惩罚函数法进行的排序结果可以满足 Deb 基于可行性规则的前两条规则, 即: 可行个体优于不可行个体; 对于两个可行个体, 选择目标函数值较小的个体. 排序得到的一些解也有可能满足第 3 条规则, 即对于两个不可行个体, 选择违反约束较小的个体.

③ $\lambda_{\mathrm{sem}}^{\min} \leqslant \lambda \leqslant \lambda_{\mathrm{sem}}^{\max}$. 一些不可行个体将优于可行个体, 即只有一部分个体满足 Deb 基于可行性规则的第 1 条规则.

④ $\lambda < \lambda_{\mathrm{sem}}^{\min}$. 所有的不可行个体均优于可行个体. 这两种方法, 仅在比较可行个体时具有相同的效果.

⑤ $\lambda < \lambda_{\min}$. 所有的不可行个体均优于可行个体, 且不可行个体根据违反约束的倒数进行排序. 这种排序得到的结果也是唯一的.

综上, 当 $\lambda > \lambda_{\max}$ 或 $\lambda < \lambda_{\min}$ 时, 排序不再依赖于惩罚系数 λ.

2.3　整体结论

由前面讨论可知, 一些结论可以概括如下:

1) 对于可行情形, 由于不牵涉到约束违反, 排序结果与 λ 无关, 两种方法具有相同的效果.

2) 对于不可行和半可行情形, 对整个种群进行排序, 当 $\lambda > \lambda_{\max}$ 或 $\lambda < \lambda_{\min}$ 时, 均可得到唯一的排序结果, 与具体取值无关. 这里的 $\lambda_{\max}$ 和 $\lambda_{\min}$ 均由当前代的种群解的情况决定.

由此可以得出, 在当前代对整个种群进行排序时, 使用一个特别大的惩罚系数 λ (如 $\lambda \gg \lambda_{\max}$) 和一个相对较小但仍大于 $\lambda_{\max}$ 的惩罚系数 λ (如 $\lambda =$

$\mathrm{floor}(\lambda_{\max})+1$) 是没有区别的, 因为它们均与 Deb 基于可行性规则排序一致; 同样地, 使用一个特别小的惩罚系数 λ (如 $\lambda \ll \lambda_{\min}$) 和一个相对较大但仍小于 $\lambda_{\min}$ 的惩罚系数 λ (如 $\lambda = \mathrm{floor}(\lambda_{\min})-1$) 也是没有区别的, 因为它们与 Deb 基于可行性规则或者相反规则组合得到的排序结果是一致的.

通常情况下, 人们会认为惩罚系数越大, 排序的结果越精确. 但是前面的分析表明, 当惩罚系数达到一定的阈值 $\lambda_{\max}$ 时, 排序将不再依赖于 λ. 通过上面的分析, 可以为具体分析设计适应性罚函数法提供有限但有效的依据.

3 实例分析

为了更好验证前面对于惩罚系数的系统分析, 这里取 $S=\{(9,3),(8,2),(10,1),(5,4),(12,0),(8,0)\}$ 作为一组处于半可行情形下的解进行验证.

根据改进的 Deb 基于可行性规则, 排序后的结果为 $S_D=\{(8,0),(12,0),(10,1),(8,2),(9,3),(5,4)\}$.

另外, 可以计算出 $\lambda_{\inf}^{\min}=-1$, $\lambda_{\inf}^{\max}=4$, $\lambda_{\mathrm{sem}}^{\min}=-2$, $\lambda_{\mathrm{sem}}^{\max}=2$, $\lambda_{\max}=\max(\lambda_{\inf}^{\max},\lambda_{\mathrm{sem}}^{\max})=4$, $\lambda_{\min}=\min(\lambda_{\inf}^{\min},\lambda_{\mathrm{sem}}^{\min})=-2$.

在 $\lambda>\lambda_{\max}$ 的情况下, 分别取 $\lambda_1=6$ 和 $\lambda_2=1\,000$, 它们对应的评价函数分别为 $L_1=f+\lambda_1 G=\{27,20,16,29,12,8\}$, $L_2=f+\lambda_2 G=\{30\,009,20\,008,10\,010,40\,005,12,8\}$, 对应的排序后索引号均为 $I=\{5,4,3,6,2,1\}$, 得到的排序结果同为 $S_p^1=\{(8,0),(12,0),(10,1),(8,2),(9,3),(5,4)\}$, 与改进的 Deb 基于可行性规则排序顺序完全相同.

类似地, 在 $\lambda<\lambda_{\min}$ 的情况下, 分别取 $\lambda_3=-3$ 和 $\lambda_4=-10\,000$, 容易得到, 它们对应的评价函数分别为 $L_3=f+\lambda_3 G=\{0,2,7,-7,12,8\}$, $L_4=f+\lambda_4 G=\{-29\,991,-19\,992,-9\,990,-39\,995,12,8\}$, 对应的排序后索引号均为 $I=\{2,3,4,1,6,5\}$, 最后得到的排序结果同为 $S_p^2=\{(5,4),(9,3),(8,2),(10,1),(8,0),(12,0)\}$. 可以看出, 所得的排序结果中, 所有不可行解均优于可行解; 对于不可行解, 与 Deb 的排序结果完全相反; 对于可行解, 由于其排序与 λ 无关, 可行解的排序仍按照目标函数值进行. 整体的排序结果唯一.

综上, 在 $\lambda>\lambda_{\max}$ 和 $\lambda<\lambda_{\min}$ 的情况下, 排序结果不受 λ 具体值的影响, 验证了前面第 2 节的分析.

4 结 论

惩罚系数的确定是使用惩罚函数法时面临的一个关键问题. 鉴于目前对其确定主要基于实验的现状, 本文尝试对其内部机理进行系统分析. 首先对 Deb 基于可行性规则进行改进; 然后以其作为参照标准, 根据规则排序结果的唯一性, 重点对不可行情形和半可行情形下其与惩罚系数的对应关系进行系统阐述, 并得到结论: 对整个种群进行排序, 当 $\lambda>\lambda_{\max}$ 或 $\lambda<\lambda_{\min}$ 时, 均可得到唯一的排序结果, 与 λ 具体取值无关. 最后的实例表明, 上面的分析是有效的, 可以为具体分析设计惩罚函数法提供有效的依据.

参考文献(References)

[1] Coello Coello C A. Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: A survey of the state of the art[J]. Computer Methods in Applied Mechanics and Engineering, 2002, 191(11/12): 1245-1287.

[2] 甘敏, 彭辉, 王勇. 多目标优化与自适应惩罚的混合约束优化进化算法[J]. 控制与决策, 2010, 25(3): 378-382. (Gan M, Peng H, Wang Y. Multiobjective optimization and adaptive penalty function based constrained optimization evolutionary algorithm[J]. Control and Decision, 2010, 25(3): 378-382.)

[3] 梁昔明, 龙文, 秦浩宇, 等. 基于种群个体可行性的约束优化进化算法[J]. 控制与决策, 2010, 25(8): 1129-1132. (Liang X M, Long W, Qin H Y, et al. Constrained optimization evolutionary algorithm based on individual feasibility of population[J]. Control and Decision, 2010, 25(8): 1129-1132.)

[4] Wang Y, Cai Z X. Combining multiobjective optimization with defferential evolution to solve constrained optimization problems[J]. IEEE Trans on Evolutionary Computation, 2012, 16(1): 117-134.

[5] Mezura-Montes E, Coello Coello C A. A simple multimembered evolution strategy to solve constrained optimization problems[J]. IEEE Trans on Evolutionary Computation, 2005, 9(1): 1-17.

[6] Deb K. An efficient constraint handling method for genetic algorithms[J]. Computer Methods in Applied Mechanics and Engineering, 2000, 186(2/3/4): 311-338.

[7] Runarsson T P, Yao X. Stochastic ranking for constrained evolutionary optimization[J]. IEEE Trans on Evolutionary Computation, 2000, 4(3): 284-294.

[8] Wang Y, Cai Z, Zhou Y, et al. An adaptive tradeoff model for constrained evolutionary optimization[J]. IEEE Trans on Evolutionary Computation, 2008, 12(1): 80-92.

(责任编辑: 李君玲)